Guía de seguridad para tu producto de Lovable

AI PM Pro · The Hero Camp · Sesión de seguridad · 10 de diciembre de 2026

Cómo usar esta guía

Esta guía explica **tres riesgos de seguridad** que aparecen a menudo en productos hechos con IA:

1. **Claves y secretos.**
2. **Datos y permisos.**
3. **Datos personales.**

Para cada riesgo encontrarás qué es, qué puede pasar, un caso real, cómo se ve en Lovable, qué hacer y un prompt para revisar tu producto.

En cuanto la recibas: léela (unos 20 minutos) y revisa tu producto con la checklist que viene después de los tres riesgos. **En la sesión (10 de diciembre):** revisaremos estos tres riesgos en directo sobre un proyecto de Lovable y resolveremos tus dudas. **Después de la sesión:** si has cambiado algo, vuelve a pasar la checklist. El Demo Day es el 12 de diciembre: no dejes los arreglos para el día 11.

Tres reglas para usar los prompts: - **Los prompts no cambian nada.** Piden a Lovable que revise tu proyecto y te explique lo que encuentra. Si Lovable propone arreglar algo, no lo aceptes todavía: primero lee el informe y decide. - **Si arreglas algo, hazlo con un Pull Request** (una propuesta de cambio que revisas antes de publicarla). Comprueba la preview (la versión de prueba) antes del merge (unir el cambio a la versión publicada), como en la sesión del 21 de noviembre. - **No apliques arreglos el día antes del Demo Day** sin haberlos probado antes.

1. Claves y secretos

Palabras que vas a ver - **Clave:** un código que permite a tu producto usar un servicio, por ejemplo la base de datos o una IA. - **Clave pública:** una clave que puede estar en la web. Por sí sola no da acceso a nada privado. - **Clave secreta:** una clave que nunca debe estar en la web. Quien la tenga puede usar el servicio en tu nombre, y pagas tú.

Qué puede pasar

Si una clave secreta se publica, otras personas la usan y la factura te llega a ti. Hay programas automáticos que buscan claves publicadas en Internet todo el tiempo: en un caso estudiado, las claves publicadas en GitHub se empezaron a usar **en menos de 5 minutos** (Unit 42, Palo Alto Networks, 2023).

Caso real

Una startup mexicana, 82.314 dólares en 48 horas (febrero de 2026). Un equipo de tres personas gastaba unos 180 dólares al mes en servicios de Google. Entre el 11 y el 12 de febrero de 2026, alguien usó su clave para generar texto e imágenes con Gemini, y la factura llegó a 82.314 dólares. Google respondió que proteger las claves es responsabilidad del cliente (The Register, 3 de marzo de 2026).

No es un caso aislado: un estudio de febrero de 2026 encontró **más de 5.000 repositorios de GitHub y unas 3.000 webs** con claves de OpenAI escritas en su código (Cyble).

Cómo se ve en Lovable

Lo comprobamos en un proyecto de Lovable el 24 de noviembre de 2026: - **Si pides poner una clave secreta en el código de la web, Lovable se niega** y propone usarla desde el servidor, que es la parte de la app que no llega al navegador. - **Si pegas una clave en el chat de Lovable, Lovable la guarda como secreto**, en un almacén protegido llamado *Secrets*. - **Lovable no puede proteger una clave que pegas en otro sitio:** GitHub, un documento, un correo, otra herramienta de IA o una captura de pantalla.

Qué hacer

1. **Guarda las claves secretas solo en *Secrets*.** No las pegues en ningún otro sitio.
2. **Pon un límite de gasto** en cada servicio que cobra por uso (OpenAI, Gemini, etc.). Algunos servicios solo te avisan cuando llegas a la cifra y no paran el gasto: comprueba qué hace cada uno y, como mínimo, activa el aviso.
3. **Si una clave secreta se publica, cámbiala en ese momento:** crea una nueva en el servicio, borra la antigua y guarda la nueva en *Secrets*.

Prompt para revisar tu producto

```
Revisa este proyecto y busca las claves de los servicios que usa (base de datos, IA, pagos, publicación). No cambies nada: solo revisa e informa.

Para cada clave que encuentres, dime:
1. De qué servicio es.
2. Si es pública (puede estar en la web) o secreta (no debe estar en la web).
3. Si está en el código que se envía al navegador del usuario o guardada en Secrets.
4. Qué podría hacer otra persona si la copia.
5. Qué me recomiendas hacer.

No incluyas las contraseñas de los usuarios, las sesiones ni los datos de prueba.
Si no encuentras ninguna clave secreta en el código del navegador, dilo claramente.

Formato de la respuesta:
- Empieza con un resumen de dos frases: si hay algo que arreglar y qué es lo más urgente.
- Después, una tabla de 5 filas como máximo, con lo más importante primero. Si hay más, agrupa en una fila las que tengan el mismo riesgo.
- Nombra cada cosa como la ve el usuario del producto (por ejemplo, "clave de OpenAI" o "formulario de registro"), sin rutas de ficheros ni nombres de variables.
- En español, con frases cortas. Explica cualquier término técnico en una frase.
```

2. Datos y permisos

Palabras que vas a ver - **Base de datos:** donde tu producto guarda la información. En tu caso, en Lovable Cloud. - **Tabla:** una parte de la base de datos con filas y columnas. Por ejemplo, una tabla de usuarios o de notas, con una fila por cada usuario o nota. - **RLS** (*Row Level Security*, seguridad por fila): las reglas que deciden qué filas puede ver, crear, cambiar o borrar cada usuario.

Qué puede pasar

Tu web usa una **clave pública** para leer y guardar datos en la base de datos. Cualquiera puede ver esa clave en el navegador, y es normal: está pensada así. **Lo que protege tus datos no es la clave, son las reglas RLS.** Si una tabla no tiene reglas, o tiene una regla que deja leerlo todo, cualquier persona con la clave pública puede leer todas sus filas: emails, mensajes, notas de tus usuarios.

Caso real

Apps hechas con Lovable con la base de datos abierta (2025). El investigador Matt Palmer publicó el 29 de mayo de 2025 que muchas apps hechas con Lovable tenían reglas RLS insuficientes: cualquier persona, sin iniciar sesión, podía leer o escribir datos de sus tablas. Según el investigador, afectaba a **más de 170 apps**. Se registró como fallo de seguridad de gravedad crítica (CVE-2025-48757, 9,3 sobre 10). **Lovable no está de acuerdo con esa clasificación:** dice que cada cliente es responsable de proteger los datos de su app.

Moltbook (enero de 2026). Una red social con una base de datos sin RLS bien configurada dejó expuestas alrededor de 1,5 millones de claves de acceso a servicios (Wiz, 2 de febrero de 2026).

Cómo se ve en Lovable

Lo comprobamos en un proyecto de Lovable el 24 de noviembre de 2026: - **Las tablas que crea Lovable tienen RLS**, con reglas para que cada usuario vea solo sus datos. - **Si tú pides una tabla que cualquiera pueda leer, Lovable la crea.** Hace lo que le pides. - **El escáner de seguridad de Lovable** (una revisión automática que busca fallos de seguridad en tu proyecto) **la marca como crítica:** "Anyone can read every record in…" (cualquiera puede leer todas las filas de…). El aviso aparece en el chat y en **More** → **Security** → **Findings**. Desde ahí, **Run checks** lanza la revisión y **Fix all** propone un arreglo.

Qué hacer

1. **Antes de publicar, abre More → Security → Findings** y lanza *Run checks*.
2. **Lee cada aviso crítico y decide.** Si una tabla tiene que ser pública, comprueba que no guarde datos personales.
3. **Si no entiendes un aviso, lanza el prompt de abajo** para que Lovable te lo explique.

Prompt para revisar tu producto

```
Revisa la base de datos de este proyecto. No cambies nada: solo revisa e informa.

Para cada tabla, dime:
1. Qué datos guarda.
2. Si tiene RLS activado.
3. Quién puede leer sus filas, quién puede crearlas, cambiarlas y borrarlas: cualquier persona sin iniciar sesión, cualquier usuario con sesión o solo el dueño de cada fila.
4. Si alguna regla permite más de lo necesario.
5. Qué me recomiendas hacer.

Ordena las tablas de mayor a menor riesgo.

Formato de la respuesta:
- Empieza con un resumen de dos frases: si hay algo que arreglar y qué es lo más urgente.
- Después, una tabla de 5 filas como máximo, con lo más importante primero. Si hay más, agrupa en una fila las que tengan el mismo riesgo.
- Nombra cada cosa como la ve el usuario del producto (por ejemplo, "clave de OpenAI" o "formulario de registro"), sin rutas de ficheros ni nombres de variables.
- En español, con frases cortas. Explica cualquier término técnico en una frase.
```

3. Datos personales

Palabras que vas a ver - **Dato personal:** cualquier dato que identifica a una persona o permite identificarla: nombre, email, teléfono, dirección, foto, DNI. - **RGPD:** la ley europea de protección de datos. Se aplica desde tu primer usuario real. - **Brecha:** cuando datos personales llegan a quien no debería tenerlos, por un error o por un ataque.

Qué puede pasar

Cada dato que pides es un dato que tienes que proteger. Si hay una brecha, **cuantos más datos tengas, más grave es.** Además, el RGPD te obliga a dos cosas: - **Pedir solo los datos que necesitas** para lo que hace tu producto. - **Avisar a la autoridad de protección de datos en un máximo de 72 horas** desde que sabes que ha habido una brecha (RGPD, artículo 33). En España, la autoridad es la AEPD.

En 2025, las autoridades europeas pusieron unos **1.200 millones de euros** en multas por incumplir el RGPD (DLA Piper, enero de 2026).

Caso real

Samsung y ChatGPT (2023). En pocas semanas, empleados de Samsung pegaron en ChatGPT código interno y notas de reuniones para que la IA les ayudara. Esa información se envió a un servicio externo, fuera del control de la empresa. Samsung limitó el uso de estas herramientas y después lo prohibió (información de la prensa coreana, recogida por The Register el 6 de abril de 2023).

Lo mismo vale para tu producto: si pegas datos reales de tus usuarios en el chat de Lovable o de otra IA para que te ayude, esos datos salen de tu producto.

Cómo se ve en Lovable

Lo comprobamos en un proyecto de Lovable el 24 de noviembre de 2026: - **Lovable se negó a escribir los datos del formulario en la consola del navegador** "por privacidad". La consola es una ventana del navegador donde la web escribe mensajes técnicos, y cualquier persona que use tu web puede abrirla. - **Pero añadió sin avisar los datos que no hacían falta:** pedimos un registro con nombre, email, teléfono, DNI y fecha de nacimiento para una app de notas, y Lovable los añadió todos. - **El escáner no lo marcó.** Solo el prompt de abajo detectó que el DNI, el teléfono y la fecha de nacimiento sobraban.

Ninguna herramienta decide por ti qué datos necesita tu producto. Es una decisión de producto.

Qué hacer

1. **Para cada dato que pide tu producto, pregúntate: ¿para qué lo uso?** Si no tienes respuesta, quítalo.
2. **No pegues datos reales de tus usuarios** en el chat de Lovable ni en otra herramienta de IA.
3. **Ten una política de privacidad** que diga qué datos pides y para qué. No hace falta que sea perfecta, pero tiene que existir.
4. **Si hay una brecha, tienes 72 horas** para avisar a la autoridad. El apartado "Si ya ocurrió" te dice por dónde empezar.

Prompt para revisar tu producto

```
Revisa qué datos personales recoge y usa este proyecto. No cambies nada: solo revisa e informa.

Dime:
1. Qué datos personales pide el producto y en qué pantalla.
2. Dónde se guardan.
3. Si alguno se escribe en los registros (logs) o en la consola del navegador.
4. Si alguno se envía a otro servicio, por ejemplo a una IA, y a cuál.
5. Si hay algún dato que el producto pide pero no necesita para funcionar.
6. Qué me recomiendas hacer.

Formato de la respuesta:
- Empieza con un resumen de dos frases: si hay algo que arreglar y qué es lo más urgente.
- Después, una tabla de 5 filas como máximo, con lo más importante primero. Si hay más, agrupa en una fila las que tengan el mismo riesgo.
- Nombra cada cosa como la ve el usuario del producto (por ejemplo, "clave de OpenAI" o "formulario de registro"), sin rutas de ficheros ni nombres de variables.
- En español, con frases cortas. Explica cualquier término técnico en una frase.
```

Checklist antes de publicar

Pásala a tu producto en cuanto recibas esta guía, y otra vez después de la sesión si has cambiado algo. Marca cada punto cuando lo hayas comprobado.

Claves y secretos

- ☐ Las claves secretas están solo en *Secrets*.
- ☐ Has lanzado el prompt 1 y dice que no hay claves secretas en el código del navegador.
- ☐ Cada servicio que cobra por uso (OpenAI, Gemini, etc.) tiene un límite de gasto o, como mínimo, un aviso de gasto.
- ☐ No hay claves secretas en GitHub, documentos, correos ni capturas de pantalla.

Datos y permisos

- ☐ Has lanzado *Run checks* en **More** → **Security** → **Findings** y has leído todos los avisos críticos.
- ☐ Si alguna tabla es pública a propósito, no guarda datos personales.
- ☐ Has lanzado el prompt 2 y cada usuario ve solo sus propios datos.

Datos personales

- ☐ Para cada dato que pide tu producto, sabes para qué lo usas. Los que no usas, los has quitado.
- ☐ Has lanzado el prompt 3 y no hay datos personales en la consola del navegador ni en los registros (los mensajes que tu producto guarda sobre lo que hace, también llamados *logs*).
- ☐ No has pegado datos reales de tus usuarios en el chat de Lovable ni en otra herramienta de IA.
- ☐ Tu producto tiene una política de privacidad que dice qué datos pide y para qué.

Configuración

La **configuración** son los ajustes de tu producto que no son la pantalla ni los datos: quién puede ver los archivos, qué páginas están publicadas, si el repositorio es público. Estos fallos no dan ningún error, por eso pasan desapercibidos. - ☐ **Archivos de tus usuarios:** si tu producto guarda archivos (fotos, documentos), pregunta a Lovable en el chat si cualquiera puede abrirlos con el enlace. Deben verlos solo las personas que tienen permiso. - ☐ **Páginas de prueba:** pregunta a Lovable si hay páginas de prueba o de administración que se puedan abrir sin iniciar sesión. Si las hay y no las usas, pide que las quite. - ☐ **Cabeceras de seguridad** (instrucciones que tu web envía al navegador para protegerse de algunos ataques): pregunta a Lovable si tu web publicada las envía. - ☐ **Repositorio de GitHub:** es privado, salvo que hayas decidido que sea público.

Si ya ocurrió: los primeros 30 minutos

Si descubres que una clave secreta se ha publicado o que otras personas han visto datos de tus usuarios, sigue estos pasos en este orden.

1. **Cambia la clave en ese momento.** Crea una nueva en el servicio, borra la antigua y guarda la nueva en *Secrets*. Hazlo antes de investigar: mientras la clave antigua funcione, alguien puede seguir usándola.
2. **Cierra el acceso a los datos.** Si una tabla se podía leer sin permiso, pide a Lovable que la proteja con RLS. Aquí no esperes al Pull Request: primero cierra el acceso y después revisa el cambio.
3. **Averigua qué se vio y desde cuándo.** ¿Qué datos estaban expuestos? ¿Desde qué fecha? Si no puedes demostrar que nadie los vio, cuenta con que alguien los vio.
4. **Si hay datos personales, avisa ya a quien lleve los temas legales** en tu empresa, o a un abogado. Tienes **72 horas** para avisar a la autoridad de protección de datos (la AEPD en España), contadas desde que lo sabes. No hace falta tener toda la información para avisar.
5. **Apunta todo con fecha y hora, y no borres nada.** Haz capturas de lo que ves, apunta cuándo te enteraste y cada cosa que haces. No borres los registros del incidente: son la prueba de lo que pasó y de lo que hiciste.

Pide ayuda a un profesional de seguridad si tu producto maneja datos de salud, datos bancarios o pagos, o si un cliente o inversor te pide una auditoría.

Aviso para el Demo Day

- **Pasa la checklist con tiempo**, no el día 11. Un arreglo sin probar puede romper el producto el día de la demo.
- **Usa datos de prueba en la demo**, no datos reales de tus usuarios.
- **No enseñes en pantalla *Secrets*, la configuración de los servicios ni el código con claves.** Si la demo se graba, la grabación se puede reenviar y cualquiera puede pararla y leer lo que sale en pantalla.

> **Para curiosos: el pipeline de la sesión**
>
> Al final de la sesión verás, durante unos 5 minutos, un *pipeline* de auditoría: varios programas de IA que, uno detrás de otro, revisan un proyecto entero y escriben un informe con los fallos de seguridad. Funciona con Claude Code dentro de VS Code, que son herramientas para trabajar con el código desde el ordenador, fuera de Lovable.
>
> **No lo necesitas para tu producto.** Con el escáner de Lovable, los tres prompts y esta checklist tienes lo necesario. Lo enseñamos para que sepas que estas revisiones se pueden automatizar si un día trabajas con el código fuera de Lovable.

Fuentes

- **Prueba en Lovable (24-nov-2026):** proyecto de prueba del curso, con capturas en el repositorio (SPEC-026). - Startup mexicana: The Register, 3-mar-2026, https://www.theregister.com/2026/03/03/gemini_api_key_82314_dollar_charge/ - Claves usadas en menos de 5 minutos: Unit 42, Palo Alto Networks, https://unit42.paloaltonetworks.com/malicious-operations-of-exposed-iam-keys-cryptojacking/ - Claves de OpenAI en repositorios y webs: Cyble, feb-2026, https://cyble.com/blog/when-ai-secrets-go-public-chatgpt/ - CVE-2025-48757: NVD, https://nvd.nist.gov/vuln/detail/CVE-2025-48757 · Matt Palmer, https://mattpalmer.io/posts/2025/05/CVE-2025-48757/ - Moltbook: Wiz, 2-feb-2026, https://www.wiz.io/blog/exposed-moltbook-database-reveals-millions-of-api-keys - Multas RGPD 2025: DLA Piper, enero de 2026, https://www.dlapiper.com/en/insights/publications/2026/01/dla-piper-gdpr-fines-and-data-breach-survey-january-2026 - Samsung y ChatGPT: The Register, 6-abr-2023 (cita a Economist Korea) - RGPD, artículo 33 (notificación de brechas en 72 horas)